Phonological Constraints and Morphological Preprocessing for Grapheme-to-Phoneme Conversion

Vera Demberg
School of Informatics
University of Edinburgh
Edinburgh, EH8 9LW, GB
v.demberg@sms.ed.ac.uk

Helmut Schmid
IMS
University of Stuttgart
D-70174 Stuttgart
schmid@ims.uni-stuttgart.de

Gregor Möhler
Speech Technologies
IBM Deutschland Entwicklung
D-71072 Böblingen
moehler@de.ibm.com

Abstract

Grapheme-to-phoneme conversion (g2p) is a core component of any text-to-speech system. We show that adding simple syllabification and stress assignment constraints, namely 'one nucleus per syllable' and 'one main stress per word', to a joint n-gram model for g2p conversion leads to a dramatic improvement in conversion accuracy.

Secondly, we assessed morphological preprocessing for g2p conversion. While morphological information has been incorporated in some past systems, its contribution has never been quantitatively assessed for German. We compare the relevance of morphological preprocessing with respect to the morphological segmentation method, training set size, the g2p conversion algorithm, and two languages, English and German.

1 Introduction

Grapheme-to-Phoneme conversion (g2p) is the task of converting a word from its spelling (e.g. "Sternanisöl", Engl: star-anise oil) to its pronunciation (/ˈʃtɛrnʔaniːsʔøːl/). Speech synthesis modules with a g2p component are used in text-to-speech (TTS) systems and can be be applied in spoken dialogue systems or speech-to-speech translation systems.

1.1 Syllabification and Stress in g2p conversion

In order to correctly synthesize a word, it is not only necessary to convert the letters into phonemes, but also to syllabify the word and to assign word stress.

The problems of word phonemization, syllabification and word stress assignment are inter-dependent. Information about the position of a syllable boundary helps grapheme-to-phoneme conversion. (Marchand and Damper, 2005) report a word error rate (WER) reduction of approx. 5 percentage points for English when the letter string is augmented with syllabification information. The same holds vice-versa: we found that WER was reduced by 50% when running our syllabifier on phonemes instead of letters (see Table 4). Finally, word stress is usually defined on syllables; in languages where word stress is assumed[1] to partly depend on syllable weight (such as German or Dutch), it is important to know where exactly the syllable boundaries are in order to correctly calculate syllable weight. For German, (Müller, 2001) show that information about stress assignment and the position of a syllable within a word improve g2p conversion.

1.2 Morphological Preprocessing

It has been argued that using morphological information is important for languages where morphology has an important influence on pronunciation, syllabification and word stress such as German, Dutch, Swedish or, to a smaller extent, also English (Sproat, 1996; Möbius, 2001; Pounder and Kommenda, 1986; Black et al., 1998; Taylor, 2005). Unfortunately, these papers do not quantify the contribution of morphological preprocessing in the task.

Important questions when considering the integration of a morphological component into a speech

[1] This issue is controversial among linguists; for an overview see (Jessen, 1998).

Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pages 96–103,
Prague, Czech Republic, June 2007.

synthesis system are 1) How large are the improvements to be gained from morphological preprocessing? 2) Must the morphological system be perfect or can performance improvements also be reached with relatively simple morphological components? and 3) How much does the benefit to be expected from explicit morphological information depend on the g2p algorithm? To determine these factors, we compared morphological segmentations based on manual morphological annotation from CELEX to two rule-based systems and several unsupervised data-based approaches. We also analysed the role of explicit morphological preprocessing on data sets of different sizes and compared its relevance with respect to a decision tree and a joint n-gram model for g2p conversion.

The paper is structured as follows: We introduce the g2p conversion model we used in section 2 and explain how we implemented the phonological constraints in section 3. Section 4 is concerned with the relation between morphology, word pronunciation, syllabification and word stress in German, and presents different sources for morphological segmentation. In section 5, we evaluate the contribution of each of the components and compare our methods to state-of-the-art systems. Section 6 summarizes our results.

2 Methods

We used a joint n-gram model for the grapheme-to-phoneme conversion task. Models of this type have previously been shown to yield very good g2p conversion results (Bisani and Ney, 2002; Galescu and Allen, 2001; Chen, 2003). Models that do not use *joint* letter-phoneme states, and therefore are not conditional on the preceding letters, but only on the actual letter and the preceding phonemes, achieved inferior results. Examples of such approaches using Hidden Markov Models are (Rentzepopoulos and Kokkinakis, 1991) (who applied the HMM to the related task of phoneme-to-grapheme conversion), (Taylor, 2005) and (Minker, 1996).

The g2p task is formulated as searching for the most probable sequence of phonemes given the orthographic form of a word. One can think of it as a tagging problem where each letter is tagged with a (possibly empty) phoneme-sequence p. In our particular implementation, the model is defined as a higher-order Hidden Markov Model, where the hidden states are a letter–phoneme-sequence pair $\langle l;p\rangle$, and the observed symbols are the letters l. The output probability of a hidden state is then equal to one, since all hidden states that do not contain the observed letter are pruned directly.

The model for grapheme-to-phoneme conversion uses the Viterbi algorithm to efficiently compute the most probable sequence $\hat{p}_1^n$ of phonemes $\hat{p}_1, \hat{p}_2, ..., \hat{p}_n$ for a given letter sequence l_1^n. The probability of a letter–phon-seq pair depends on the *k* preceding letter–phon-seq pairs. Dummy states '#' are appended at both ends of each word to indicate the word boundary and to ensure that all conditional probabilities are well-defined.

$$\hat{p}_1^n = \arg\max_{p_1^n} \prod_{i=1}^{n+1} P(\langle l;p\rangle_i \,|\, \langle l;p\rangle_{i-k}^{i-1})$$

In an integrated model where g2p conversion, syllabification and word stress assignment are all performed at the same time, a state additionally contains a syllable boundary flag b and a stress flag a, yielding $\langle l;p;b;a\rangle_i$.

As an alternative architecture, we also designed a modular system that comprises one component for syllabification and one for word stress assignment. The model for syllabification computes the most probable sequence $\hat{b}_1^n$ of syllable boundary-tags $\hat{b}_1$, $\hat{b}_2, ..., \hat{b}_n$ for a given letter sequence l_1^n.

$$\hat{b}_1^n = \arg\max_{b_1^n} \prod_{i=1}^{n+1} P(\langle l;b\rangle_i \,|\, \langle l;b\rangle_{i-k}^{i-1})$$

The stress assignment model works on syllables. It computes the most probable sequence $\hat{a}_1^n$ of word accent-tags $\hat{a}_1, \hat{a}_2, ..., \hat{a}_n$ for a given syllable sequence syl_1^n.

$$\hat{a}_1^n = \arg\max_{a_1^n} \prod_{i=1}^{n+1} P(\langle syl;a\rangle_i \,|\, \langle syl;a\rangle_{i-k}^{i-1})$$

2.1 Smoothing

Because of major data sparseness problems, smoothing is an important issue, in particular for the stress model which is based on syllable–stress-tag pairs. Performance varied by up to 20% in function of the smoothing algorithm chosen. Best results were obtained when using a variant of Modified Kneser-Ney Smoothing[2] (Chen and Goodman, 1996).

[2] For a formal definition, see(Demberg, 2006).

2.2 Pruning

In the g2p-model, each letter can on average map onto one of 12 alternative phoneme-sequences. When working with 5-grams[3], there are about 12^5 = 250,000 state sequences. To improve time and space efficiency, we implemented a simple pruning strategy that only considers the t best states at any moment in time. With a threshold of $t = 15$, about 120 words are processed per minute on a 1.5GHz machine. Conversion quality is only marginally worse than when the whole search space is calculated.

Running time for English is faster, because the average number of candidate phonemes for each letter is lower. We measured running time (including training and the actual g2p conversion in 10-fold cross validation) for a Perl implementation of our algorithm on the English NetTalk corpus (20,008 words) on an Intel Pentium 4, 3.0 GHz machine. Running time was less than 1h for each of the following three test conditions: c1) g2p conversion only, c2) syllabification first, then g2p conversion, c3) simultaneous g2p conversion and syllabification, given perfect syllable boundary input, c4) simultaneous g2p conversion and syllabification when correct syllabification is not available beforehand. This is much faster than the times for Pronunciation by Analogy (PbA) (Marchand and Damper, 2005) on the same corpus. Marchand and Damper reported a processing time of several hours for c4), two days for c2) and several days for c3).

2.3 Alignment

Our current implementation of the joint n-gram model is not integrated with an automatic alignment procedure. We therefore first aligned letters and phonemes in a separate, semi-automatic step. Each letter was aligned with zero to two phonemes and, in the integrated model, zero or one syllable boundaries and stress markers.

3 Integration of Phonological Constraints

When analysing the results from the model that does g2p conversion, syllabification and stress assignment in a single step, we found that a large proportion of the errors was due to the violation of basic phonological constraints.

Some syllables had no syllable nucleus, while others contained several vowels. The reason for the errors is that German syllables can be very long and therefore sparse, often causing the model to back-off to smaller contexts. If the context is too small to cover the syllable, the model cannot decide whether the current syllable contains a nucleus.

In stress assignment, this problem is even worse: the context window rarely covers the whole word. The algorithm does not know whether it already assigned a word stress outside the context window. This leads to a high error rate with 15-20% of incorrectly stressed words. Thereof, 37% have more than one main stress, about 27% are not assigned any stress and 36% are stressed in the wrong position. This means that we can hope to reduce the errors by almost 2/3 by using phonological constraints.

Word stress assignment is a difficult problem in German because the underlying processes involve some deeper morphological knowledge which is not available to the simple model. In complex words, stress mainly depends on morphological structure (i.e. on the compositionality of compounds and on the stressing status of affixes). Word stress in simplex words is assumed to depend on the syllable position within the word stem and on syllable weight. The current language-independent approach does not model these processes, but only captures some of its statistics.

Simple constraints can help to overcome the problem of lacking context by explicitly requiring that every syllable must have exactly one syllable nucleus and that every word must have exactly one syllable receiving primary stress.

3.1 Implementation

Our goal is to find the most probable syllabified and stressed phonemization of a word that does not violate the constraints. We tried two different approaches to enforce the constraints.

In the first variant (v1), we modified the probability model to enforce the constraints. Each state now corresponds to a sequence of 4-tuples consisting of a letter l, a phoneme sequence p, a syllable boundary tag b, an accent tag a (as before) plus two

[3]There is a trade-off between long context windows which capture the context accurately and data sparseness issues. The optimal value k for the context window size depends on the source language (existence of multiletter graphemes, complexity of syllables etc.).

new flags A and N which indicate whether an accent/nucleus precedes or not. The A and N flags of the new state are a function of its accent and syllable boundary tag and the A and N flag of the preceding state. They split each state into four new states. The new transition probabilities are defined as:

$$P(\langle l; p; b; a\rangle_i \,|\, \langle l; p; b; a\rangle_{i-k}^{i-1}, A, N)$$

The probability is 0 if the transition violates a constraint, e.g., when the A flag is set and a_i indicates another accent.

A positive side effect of the syllable flag is that it stores separate phonemization probabilities for consonants in the syllable onset vs. consonants in the coda. The flag in the onset is 0 since the nucleus has not yet been encountered, whereas it is set to 1 in the coda. In German, this can e.g. help in for syllable-final devoicing of voiced stops and fricatives.

The increase in the number of states aggravates sparse-data problems. Therefore, we implemented another variant (v2) which uses the same set of states (with A and N flags), but with the transition probabilities of the original model, which did not enforce the constraints. Instead, we modified the Viterbi algorithm to eliminate the invalid transitions: For example, a transition from a state with the A flag set to a state where a_i introduces a second stress, is always ignored. On small data sets, better results were achieved with v2 (see Table 5).

4 Morphological Preprocessing

In German, information about morphological boundaries is needed to correctly insert glottal stops [ʔ] in complex words, to determine irregular pronunciation of affixes (*v* is pronounced [v] in *vertikal* but [f] in *ver+ticker+n*, and the suffix syllable *heit* is not stressed although superheavy and word final) and to disambiguate letters (e.g. *e* is always pronounced /ə/ when occurring in inflectional suffixes). Vowel length and quality has been argued to also depend on morphological structure (Pounder and Kommenda, 1986). Furthermore, morphological boundaries overrun default syllabification rules, such as the maximum onset principle.

Applying default syllabification to the word "Sternanisöl" would result in a syllabification into `Ster-na-ni-söl` (and subsequent phonemization to something like /ʃtɛʀˈnaːnizøːl/) instead of `Stern-a-nis-öl` (/ˈʃtɛʀnʔaniːsʔøːl/). Syllabification in turn affects phonemization since voiced fricatives and stops are devoiced in syllable-final position. Morphological information also helps for graphemic parsing of words such as "Röschen" (Engl: little rose) where the morphological boundary between *Rös* and *chen* causes the string *sch* to be transcribed to /sç/ instead of /ʃ/. Similar ambiguities can arise for all other sounds that are represented by several letters in orthography (e.g. doubled consonants, diphtongs, *ie*, *ph*, *th*), and is also valid for English. Finally, morphological information is also crucial to determine word stress in morphologically complex words.

4.1 Methods for Morphological Segmentation

Good segmentation performance on arbitrary words is hard to achieve. We compared several approaches with different amounts of built-in knowledge. The morphological information is encoded in the letter string, where different digits represent different kinds of morphological boundaries (prefixes, stems, derivational and inflectional suffixes).

Manual Annotation from CELEX

To determine the upper bound of what can be achieved when exploiting perfect morphological information, we extracted morphological boundaries and boundary types from the CELEX database.

The manual annotation is not perfect as it contains some errors and many cases where words are not decomposed entirely. The words tagged [F] for "lexicalized inflection", e.g. *gedrängt* (past participle of *drängen*, Engl: push) were decomposed semi-automatically for the purpose of this evaluation. As expected, annotating words with CELEX morphological segmentation yielded the best g2p conversion results. Manual annotation is only available for a small number of words. Therefore, only automatically annotated morphological information can scale up to real applications.

Rule-based Systems

The traditional approach is to use large morpheme lexica and a set of rules that segment words into affixes and stems. Drawbacks of using such a system are the high development costs, limited coverage and problems with ambiguity resolution between alternative analyses of a word.

The two rule-based systems we evaluated, the ETI[4] morphological system and SMOR[5] (Schmid et al., 2004), are both high-quality systems with large lexica that have been developed over several years. Their performance results can help to estimate what can realistically be expected from an automatic segmentation system. Both of the rule-based systems achieved an F-score of approx. 80% morphological boundaries correct with respect to CELEX manual annotation.

Unsupervised Morphological Systems

Most attractive among automatic systems are methods that use unsupervised learning, because these require neither an expert linguist to build large rule-sets and lexica nor large manually annotated word lists, but only large amounts of tokenized text, which can be acquired e.g. from the internet. Unsupervised methods are in principle[6] language-independent, and can therefore easily be applied to other languages.

We compared four different state-of-the-art unsupervised systems for morphological decomposition (cf. (Demberg, 2006; Demberg, 2007)). The algorithms were trained on a German newspaper corpus (taz), containing about 240 million words. The same algorithms have previously been shown to help a speech recognition task (Kurimo et al., 2006).

5 Experimental Evaluations

5.1 Training Set and Test Set Design

The German corpus used in these experiments is CELEX (German Linguistic User Guide, 1995). CELEX contains a phonemic representation of each word, syllable boundaries and word stress information. Furthermore, it contains manually verified morphological boundaries.

Our training set contains approx. 240,000 words and the test set consists of 12,326 words. The test set is designed such that word stems in training and test sets are disjoint, i.e. the inflections of a certain stem are either all in the training set or all in the test set. Stem overlap between training and test set only occurs in compounds and derivations. If a simple random splitting (90% for training set, 10% for test set) is used on inflected corpora, results are much better: Word error rates (WER) are about 60% lower when the set of stems in training and test set are not disjoint. The same effect can also be observed for the syllabification task (see Table 4).

5.2 Results for the Joint n-gram Model

The joint n-gram model is language-independent. An aligned corpus with words and their pronunciations is needed, but no further adaptation is required.

Table 1 shows the performance of our model in comparison to alternative approaches on the German and English versions of the CELEX corpus, the English NetTalk corpus, the English Teacher's Word Book (TWB) corpus, the English beep corpus and the French Brulex corpus. The joint n-gram model performs significantly better than the decision tree (essentially based on (Lucassen and Mercer, 1984)), and achieves scores comparable to the Pronunciation by Analogy (PbA) algorithm (Marchand and Damper, 2005). For the Nettalk data, we also compared the influence of syllable boundary annotation from a) automatically learnt and b) manually annotated syllabification information on phoneme accuracy. Automatic syllabification for our model integrated phonological constraints (as described in section 3.1), and therefore led to an improvement in phoneme accuracy, while the word error rate increased for the PbA approach, which does not incorporate such constraints.

(Chen, 2003) also used a joint n-gram model. The two approaches differ in that Chen uses small chunks ($\langle (l : |0..1|) : (p : |0..1|) \rangle$ pairs only) and iteratively optimizes letter-phoneme alignment during training. Chen smoothes higher-order Markov Models with Gaussian Priors and implements additional language modelling such as consonant doubling.

[4]Eloquent Technology, Inc. (ETI) TTS system. http://www.mindspring.com/~ssshp/ssshp_cd/ss_eloq.htm

[5]The lexicon used by SMOR, IMSLEX, contains morphologically complex entries, which leads to high precision and low recall. The results reported here refer to a version of SMOR, where the lexicon entries were decomposed using a rather naïve high-recall segmentation method. SMOR itself does not disambiguate morphological analyses of a word. Our version used transition weights learnt from CELEX morphological annotation. For more details refer to (Demberg, 2006).

[6]Most systems make some assumptions about the underlying morphological system, for instance that morphology is a concatenative process, that stems have a certain minimal length or that prefixing and suffixing are the most relevant phenomena.

corpus	size	jnt n-gr	PbA	Chen	dec.tree
G - CELEX	230k	7.5%			15.0%
E - Nettalk	20k	35.4%	34.65%	34.6%	
a) auto.syll		35.3%	35.2%		
b) man.syll		29.4%	28.3%		
E - TWB	18k	28.5%	28.2%		
E - beep	200k	14.3%	13.3%		
E - CELEX	100k	23.7%			31.7%
F - Brulex	27k	10.9%			

Table 1: Word error rates for different g2p conversion algorithms. Constraints were only used in the E-Nettalk auto. syll condition.

5.3 Benefit of Integrating Constraints

The accuracy improvements achieved by integrating the constraints (see Table 2) are highly statistically significant. The numbers for conditions "G-syllab.+stress+g2p" and "E-syllab.+g2p" in Table 2 differ from the numbers for "G-CELEX" and "E-Nettalk" in Table 1 because phoneme conversion errors, syllabification errors and stress assignment errors are all counted towards word error rates reported in Table 2.

Word error rate in the combined g2p-syllable-stress model was reduced from 21.5% to 13.7%. For the separate tasks, we observed similar effects: The word error rate for inserting syllable boundaries was reduced from 3.48% to 3.1% on letters and from 1.84% to 1.53% on phonemes. Most significantly, word error rate was decreased from 30.9% to 9.9% for word stress assignment on graphemes.

We also found similarly important improvements when applying the syllabification constraint to English grapheme-to-phoneme conversion and syllabification. This suggests that our findings are not specific to German but that this kind of general constraints can be beneficial for a range of languages.

	no constr.	constraint(s)
G - syllab.+stress+g2p	21.5%	13.7%
G - syllab. on letters	3.5%	3.1%
G - syllab. on phonemes	1.84%	1.53%
G - stress assignm. on letters	30.9%	9.9%
E - syllab.+g2p	40.5%	37.5%
E - syllab. on phonemes	12.7%	8.8%

Table 2: Improving performance on g2p conversion, syllabification and stress assignment through the introduction of constraints. The table shows word error rates for German CELEX (G) and English NetTalk (E).

5.4 Modularity

Modularity is an advantage if the individual components are more specialized to their task (e.g. by applying a particular level of description of the problem, or by incorporating some additional source of knowledge).In a modular system, one component can easily be substituted by another – for example, if a better way of doing stress assignment in German was found. On the other hand, keeping everything in one module for strongly inter-dependent tasks (such as determining word stress and phonemization) allows us to simultaneously optimize for the best combination of phonemes and stress.

Best results were obtained from the joint n-gram model that does syllabification, stress assignment and g2p conversion all in a single step and integrates phonological constraints for syllabification and word stress (WER = 14.4% using method v1, WER = 13.7% using method v2). If the modular architecture is chosen, best results are obtained when g2p conversion is done before syllabification and stress assignment (15.2% WER), whereas doing syllabification and stress assignment first and then g2p conversion leads to a WER of 16.6%. We can conclude from this finding that an integrated approach is superior to a pipeline architecture for strongly interdependent tasks such as these.

5.5 The Contribution of Morphological Preprocessing

A statistically significant (according to a two-tailed t-test) improvement in g2p conversion accuracy (from 13.7% WER to 13.2% WER) was obtained with the manually annotated morphological boundaries from CELEX. The segmentation from both of the rule-based systems (ETI and SMOR) also resulted in an accuracy increase with respect to the baseline (13.6% WER), which is not annotated with morphological boundaries.

Among the unsupervised systems, best results[7] on the g2p task with morphological annotation were obtained with the RePortS system (Keshava and Pitler, 2006). But none of the segmentations led to an error reduction when compared to a baseline that used no morphological information (see Table 3). Word error rate even increased when the quality of the

[7]For all results refer to (Demberg, 2006).

	Precis.	Recall	F-Meas.	WER
RePortS (unsuperv.)	71.1%	50.7%	59.2%	15.1%
no morphology				13.7%
SMOR (rule-based)	87.1%	80.4%	83.6%	
ETI (rule-based)	75.4%	84.1%	79.5%	13.6%
CELEX (manual)	100%	100%	100%	13.2%

Table 3: Systems evaluation on German CELEX manual annotation and on the g2p task using a joint n-gram model. WERs refer to implementation v2.

morphological segmentation was too low (the unsupervised algorithms achieved 52%-62% F-measure with respect to CELEX manual annotation).

Table 4 shows that high-quality morphological information can also significantly improve performance on a syllabification task for German. We used the syllabifier described in (Schmid et al., 2005), which works similar to the joint n-gram model used for g2p conversion. Just as for g2p conversion, we found a significant accuracy improvement when using the manually annotated data, a smaller improvement for using data from the rule-based morphological system, and no improvement when using segmentations from an unsupervised algorithm. Syllabification works best when performed on phonemes, because syllables are phonological units and therefore can be determined most easily in terms of phonological entities such as phonemes.

Whether morphological segmentation is worth the effort depends on many factors such as training set size, the g2p algorithm and the language considered.

	disj. stems	random
RePortS (unsupervised morph.)	4.95%	
no morphology	3.10%	0.72%
ETI (rule-based morph.)	2.63%	
CELEX (manual annot.)	1.91%	0.53%
on phonemes	1.53%	0.18%

Table 4: Word error rates (WER) for syllabification with a joint n-gram model for two different training and test set designs (see Section 5.1).

Morphology for Data Sparseness Reduction

Probably the most important aspect of morphological segmentation information is that it can help to resolve data sparseness issues. Because of the additional knowledge given to the system through the morphological information, similarly-behaving letter sequences can be grouped more effectively.

Therefore, we hypothesized that morphological information is most beneficial in situations where the training corpus is rather small. Our findings confirm this expectation, as the relative error reduction through morphological annotation for a training corpus of 9,600 words is 6.67%, while it is only 3.65% for a 240,000-word training corpus.

In our implementation, the stress flags and syllable flags we use to enforce the phonological constraints increase data sparseness. We found v2 (the implementation that uses the states without stress and syllable flags and enforces the constraints by eliminating invalid transitions, cf. section 3.1) to outperform the integrated version, v1, and more significantly in the case of more severe data sparseness. The only condition when we found v1 to perform better than v2 was with a large data set and additional data sparseness reduction through morphological annotation, as in section 4 (see Table 5).

WER: designs	v1		v2	
data set size	240k	9.6k	240k	9.6k
no morph.	14.4%	32.3%	**13.7%**	25.5%
CELEX	**12.5%**	29%	13.2%	23.8%

Table 5: The interactions of constraints in training and different levels of data sparseness.

g2p Conversion Algorithms

The benefit of using morphological preprocessing is also affected by the algorithm that is used for g2p conversion. Therefore, we also evaluated the relative improvement of morphological annotation when using a decision tree for g2p conversion.

Decision trees were one of the first data-based approaches to g2p and are still widely used (Kienappel and Kneser, 2001; Black et al., 1998). The tree's efficiency and ability for generalization largely depends on pruning and the choice of possible questions. In our implementation, the decision tree can ask about letters within a context window of five back and five ahead, about five phonemes back and groups of letters (e.g. consonants vs. vowels).

Both the decision tree and the joint n-gram model convert graphemes to phonemes, insert syllable boundaries and assign word stress in a single step (marked as "WER-ss" in Table 6. The implementation of the joint n-gram model incorporates the phonological constraints described in section 3 ("WER-ss+"). Our main finding is that the joint n-gram model profits less from morphological annotation. Without the constraints, the performance difference is smaller: the joint n-gram model then achieves a word error rate of 21.5% on the no-morphology-condition.

In very recent work, (Demberg, 2007) developed an unsupervised algorithm (f-meas: 68%; an extension of RePortS) whose segmentations improve g2p when using a the decision tree (PER: 3.45%).

	decision tree		joint n-gram	
	PER	WER-ss	PER	WER-ss$^+$
RePortS	3.83%	28.3%		15.1%
no morph.	3.63%	26.59%	2.52%	13.7%
ETI	2.8%	21.13%	2.53%	13.6%
CELEX	2.64%	21.64%	2.36%	13.2%

Table 6: The effect of morphological preprocessing on phoneme error rates (PER) and word error rates (WER) in grapheme-to-phoneme conversion.

Morphology for other Languages

We also investigated the effect of morphological information on g2p conversion and syllabification in English, using manually annotated morphological boundaries from CELEX and the automatic unsupervised RePortS system which achieves an F-score of about 77% for English. The cases where morphological information affects word pronunciation are relatively few in comparison to German, therefore the overall effect is rather weak and we did not even find improvements with perfect boundaries.

6 Conclusions

Our results confirm that the integration of phonological constraints 'one nucleus per syllable' and 'one main stress per word' can significantly boost accuracy for g2p conversion in German and English. We implemented the constraints using a joint n-gram model for g2p conversion, which is language-independent and well-suited to the g2p task.

We systematically evaluated the benefit to be gained from morphological preprocessing on g2p conversion and syllabification. We found that morphological segmentations from rule-based systems led to some improvement. But the magnitude of the accuracy improvement strongly depends on the g2p algorithm and on training set size. State-of-the-art unsupervised morphological systems do not yet yield sufficiently good segmentations to help the task, if a good conversion algorithm is used: Low quality segmentation even led to higher error rates.

Acknowledgments

We would like to thank Hinrich Schütze, Frank Keller and the ACL reviewers for valuable comments and discussion.
The first author was supported by Evangelisches Studienwerk e.V. Villigst.

References

M. Bisani and H. Ney. 2002. Investigations on joint multigram models for grapheme-to-phoneme conversion. In *ICSLP*.

A. Black, K. Lenzo, and V. Pagel. 1998. Issues in building general letter to sound rules. In *3. ESCA on Speech Synthesis*.

SF Chen and J Goodman. 1996. An empirical study of smoothing techniques for language modeling. In *Proc. of ACL*.

S. F. Chen. 2003. Conditional and joint models for grapheme-to-phoneme conversion. In *Eurospeech*.

V. Demberg. 2006. Letter-to-phoneme conversion for a German TTS-System. Master's thesis. *IMS, Univ. of Stuttgart*.

V. Demberg. 2007. A language-independent unsupervised model for morphological segmentation. In *Proc. of ACL-07*.

L. Galescu and J. Allen. 2001. Bi-directional conversion between graphemes and phonemes using a joint n-gram model. In *Proc. of the 4th ISCA Workshop on Speech Synthesis*.

CELEX German Linguistic User Guide, 1995. *Center for Lexical Information*. Max-Planck-Institut for Psycholinguistics, Nijmegen.

M. Jessen, 1998. *Word Prosodic Systems in the Languages of Europe*. Mouton de Gruyter: Berlin.

S. Keshava and E. Pitler. 2006. A simpler, intuitive approach to morpheme induction. In *Proceedings of 2nd Pascal Challenges Workshop*, pages 31–35, Venice, Italy.

A. K. Kienappel and R. Kneser. 2001. Designing very compact decision trees for grapheme-to-phoneme transcription. In *Eurospeech*, Scandinavia.

M. Kurimo, M. Creutz, M. Varjokallio, E. Arisoy, and M. Saraclar. 2006. Unsupervsied segmentation of words into morphemes – Challenge 2005: An introduction and evaluation report. In *Proc. of 2nd Pascal Challenges Workshop*, Italy.

J. Lucassen and R. Mercer. 1984. An information theoretic approach to the automatic determination of phonemic baseforms. In *ICASSP 9*.

Y. Marchand and R. I. Damper. 2005. Can syllabification improve pronunciation by analogy of English? *Natural Language Engineering*.

W. Minker. 1996. Grapheme-to-phoneme conversion - an approach based on hidden markov models.

B. Möbius. 2001. *German and Multilingual Speech Synthesis*. phonetic AIMS, Arbeitspapiere des Instituts für Maschinelle Spachverarbeitung.

K. Müller. 2001. Automatic detection of syllable boundaries combining the advantages of treebank and bracketed corpora training. In *Proceedings of ACL*, pages 402–409.

A. Pounder and M. Kommenda. 1986. Morphological analysis for a German text-to-speech system. In *COLING 1986*.

P.A. Rentzepopoulos and G.K. Kokkinakis. 1991. Phoneme to grapheme conversion using HMM. In *Eurospeech*.

H. Schmid, A. Fitschen, and U. Heid. 2004. SMOR: A German computational morphology covering derivation, composition and inflection. In *Proc. of LREC*.

H. Schmid, B. Möbius, and J. Weidenkaff. 2005. Tagging syllable boundaries with hidden Markov models. IMS, unpub.

R. Sproat. 1996. Multilingual text analysis for text-to-speech synthesis. In *Proc. ICSLP '96*, Philadelphia, PA.

P. Taylor. 2005. Hidden Markov models for grapheme to phoneme conversion. In *INTERSPEECH*.